

An Approach to CMG Steering Using Feedback Linearization (E)

by

John Dzielski¹, Edward Bergmann², Joseph Paradiso², Derek Rowell³, David Wormley³

ABSTRACT

This paper presents an approach for controlling spacecraft equipped with control moment gyroscopes. A technique from feedback linearization theory is used to transform the original nonlinear problem to an equivalent linear form without approximating assumptions. In this form, the spacecraft dynamics appear linearly, and are decoupled from redundancy in the system of gyroscopes. An general approach to distributing control effort among the available actuators is described which includes provisions for redistribution of rotors, explicit bounds in gimbal rates, and guaranteed operation at and near singular configurations. A particular algorithm is developed for systems of double-gimbal devices, and demonstrated in two examples for which existing approaches fail to give adequate performance.

(NASA-CR-183172) AN APPROACH TO CMG
STEERING USING FEEDBACK LINEARIZATION
(Applied Research Lab.) 35 p CSCI 09B

N88-30369

Unclas
G3/63 0167491

¹Applied Research Laboratory - Pennsylvania State University

²C. S. Draper Laboratory

³Massachusetts Institute of Technology

(*) This research conducted under NASA contract NAS9-17560

Introduction

There are a variety of actuators available to generate torques on a spacecraft for the purpose of attitude control. While propulsive actuators such as reaction control jets are probably the most common, there is another class of actuators which operate by transferring momentum to and from the spacecraft via a momentum "storage" system. A control moment gyroscope (CMG) is constructed with a rotor that spins at a constant speed and is mounted in a gimbal system which can reorient the flywheel. CMGs can be used to apply torques to a spacecraft by rotating the angular momentum vectors of the individual rotors. Moving the individual rotors changes the net angular momentum of the group of CMGs. In the absence of external torques, momentum conservation implies a corresponding change in the vehicle momentum, or equivalently, the angular rate of the vehicle. Because CMGs are capable of storing and smoothly manipulating large quantities of momentum over long periods of time, they are often favored in applications requiring precision pointing and the attitude control of large, long duration spacecraft.

In any particular application, control torques are generated by manipulating the momentum stored in a group of CMGs. The net stored momentum is given by a function of the form

$$h = h_{\text{cmg}}(\theta) \quad (1.1)$$

where $h \in \mathbb{R}^3$ is the net CMG momentum vector in a spacecraft-fixed reference frame, $\theta \in \mathbb{R}^m$ denotes the set of gimbal displacements, and $h_{\text{cmg}} : \mathbb{R}^m \rightarrow \mathbb{R}^3$ maps the gimbal angles into the momentum space based on the types of CMGs involved and the geometry of the mounting configuration. This equation can be differentiated with respect to time in a spacecraft-fixed reference frame to yield

$$\tau = \dot{h} = \omega \times h + D[h_{\text{cmg}}(\theta)] \dot{\theta} \quad (1.2)$$

where ω is the angular rate of the spacecraft, $D[h_{\text{cmg}}(\theta)]$ denotes the derivative (Jacobian) of the mapping $h_{\text{cmg}}(\theta)$, and $-\tau$ is the torque applied to the spacecraft due to the CMG gimbal motion. The torque is typically specified by an autopilot or a feedback control law.

In general, the number of degrees-of-freedom in the CMG system, m , is greater than the number required to provide three-axis control of the spacecraft. The problem that naturally arises is how to choose the gimbal rates, $\dot{\theta}$, to achieve a specified torque, τ . This so-called *steering problem* is complicated by the inherent nonlinearity in these devices as well as the need to manage the redundancy in a CMG configuration to avoid geometric singularities. Singularities occur when the state of the CMG system is such that the output torque, τ , cannot be specified arbitrarily due to a drop in rank of the derivative in (1.2). The loss of rank corresponds to a loss of three-axis controllability of the spacecraft.

There have been a number of methods proposed for solving the CMG steering problem. Many of these methods are based on the idea of maximizing a configuration dependent objective function subject to the constraint that a specified torque is produced. The objective function is chosen such that its value reflects the desirability of a given configuration, and is usually expressed as a function of the gimbal angles. In [2], the objective function used by the steering algorithm is the inverse of the *gain* of a double-gimbal CMG configuration. The gain is defined to be the product of the singular values of $D[h_{\text{cmg}}(\theta)]$, and singular configurations correspond to a gain of zero since they are associated with a drop in rank of this matrix. The gain function was considered in [3] and found to be ineffective for single gimbal devices due to the fact that it often cannot predict an impending singularity early enough to avoid it. In addition, it has been demonstrated that trajectories following local optima of the gain can lead directly into singularities [4]. In the referenced article, the authors use a table look-up to identify the configuration of globally optimal gain for each value of momentum for a specific mounting of four single-gimbal CMGs. This approach is not adaptable to configuration changes, and requires storing a very large data file.

To attain a margin of global effectiveness, other authors have investigated steering laws which either explicitly or implicitly locate singularities in the gimbal space and rotate the gimbal angles to avoid them. The steering laws proposed in [3,5,6] steer the CMGs away from configurations that are known to be singular. The approach of [5] is to compute the sets of gimbal angles which are singular with respect to the requested torque vector and rotate the gimbals to avoid them. In other words, gimbal angles are found for each CMG such that the CMG cannot instantaneously project momentum in the desired direction. The steering law then chooses rates that steer away from these gimbal states. The steering law in [6] exploits knowledge of a very specific mounting configuration to design an algorithm which implicitly avoids singularities.

The method of [3,11] is probably the most adaptable approach in that changes in hardware and CMG configuration are easily accommodated, and the objective function can be changed to reflect the most urgent of the CMG configuration priorities. In addition, the algorithm can be used to combine CMGs with jets to maintain vehicle control outside the torque envelope of the CMG system alone. Another advantage of this approach is that it explicitly incorporates limits on the maximum rates achievable by the gimbal drive motors. All other approaches that include this limitation incorporate it by scaling after a set of gimbal rates has been computed; thus preserving the direction chosen by the optimization, but violating the requirement that a given torque be produced.

In [1], a number of algorithms are investigated for the control of single-gimbal CMGs. One of these methods is the singularity robust inverse method [7], which attempts to solve (1.2)

by defining a vector consisting of the error in the torque produced by a choice of gimbal rates concatenated with the set of gimbal rates. A solution is found by minimizing a weighted Euclidean norm of this vector. Near singular states, this algorithm maintains reasonable gimbal rates by sacrificing torque in the direction associated with the singularity. While the singularity robust inverse yields a steering algorithm that continues to function at singular states, the algorithm itself makes no effort to avoid these undesirable configurations.

Even though CMGs have been used in a number of applications, they have not found widespread use. Single-gimbal CMGs have been used on balloon born platforms [4], in scissored pairs on early versions of NASA's Manned Maneuvering Unit, and have recently been installed on the Soviet Union's MIR space station [8]. Double-gimbal CMGs were used in NASA's Skylab [9] and are proposed for use on the planned NASA space station [10]. The principle difficulty in using CMGs for spacecraft attitude control is finding reliable solutions to the steering problem. One reason for this difficulty is that the steering problem does not consider the long term behavior of the system, but uses only the current state and torque request to compute gimbal rates. In other words, steering laws are local in nature and do not consider the effects of current decisions on the future. As a result, there is no steering law which can guarantee that it will always keep the CMG configuration out of singular states.

A useful approach to CMG steering must provide a framework for exploiting redundancy to maintain CMG authority, behave in a sensible way near singular configurations, and be able to use the CMGs at their physical limitations when necessary. Though all steering algorithms attempt to redistribute individual rotors to more desirable configurations, [3,11] is the only published algorithm which treats gimbal limitations as a real constraint. With the exception of the singularity robust inverse method which does not redistribute gimbal angles [1], none of the published algorithms address the numerical difficulties that will occur near a singularity. This paper presents a general framework for the control of spacecraft with CMGs that considers each of the three requirements listed above.

The approach presented in this paper is part of an investigation into applying the feedback linearization theory of [12,13] to the control of CMG equipped spacecraft [14]. Feedback linearization theory has been applied to several problems in spacecraft control. In [15,16], the theory is used to study optimal attitude maneuvers for torque controlled spacecraft, and spacecraft equipped with reaction wheels, respectively. Although the attitude control problems considered in both of these papers are quite similar, neither provides a realistic treatment when redundant actuators are included. In addition to the fact that redundancy is an important characteristic of CMG control systems, reaction wheel systems do not exhibit geometric singularities like CMG systems.

Applying the feedback linearization theory to the spacecraft control problem with CMGs, a transformation is obtained which simultaneously linearizes the dynamics of the spacecraft and decouples the redundant dynamics in the actuation system from the vehicle dynamics. This transformation is used to propose a class of steering algorithms in which a desired set of gimbal rates is computed, and then a normed approximation problem is solved to obtain a *best* match to the desired rates. This approximation problem is easily solved even when explicit bounds are included on the allowable gimbal rates.

Using the approach to CMG steering developed here, the problem of designing a steering law is reduced to determining an algorithm for computing gimbal rates which move the CMG rotors toward desired configurations. This is conceptually simpler than the conventional approach of determining a function whose derivative can be used to compute desired rates. Another advantage is that once a basic algorithm has been implemented, it is possible to add decision rules which can recognize and correct for known failure modes, or to design an algorithm which is entirely rule based. Such an *expert system* type of approach was employed with Skylab where much of the problem resolution was performed manually. A modification to the basic algorithm is described that guarantees operation at and near singular states.

In the following two sections a model for a CMG equipped spacecraft is described, and a linearizing transformation is obtained. Because the spacecraft dynamics appear to be linear and independent of a subset of the CMGs, linear control laws can be used to control the spacecraft attitude. Because the linearization obtained here is not an approximation, these control laws guarantee acceptable performance even for large rotations. A method for distributing gimbal rates among a group of CMG gimbals is then stated as a normed approximation problem requiring a set of desired rates for the gimbals. A particular algorithm for desired rate computation is then proposed for double-gimbal CMGs. For the parallel mounting configuration proposed for the NASA space station, two examples are presented in which published algorithms are known to fail. Note that because the geometric problem of singularity avoidance is considerably different for systems consisting of single-gimbal CMGs, the particular algorithm developed here for double-gimbal CMGs is not applicable to these devices. Also, the attitude control problem is not discussed in this paper.

Model Derivation for a CMG Equipped Spacecraft

In this section, a mathematical model is developed for a CMG equipped spacecraft. The vehicle attitude is represented by a quaternion, and this section includes a brief introduction to quaternions and quaternion algebra. It will be shown in the following section that the attitude variable to be controlled can be chosen arbitrarily once a specific vehicle model is defined.

Following the introduction to quaternions, a kinematical model for a double-gimbal CMG is presented. This kinematical model is then incorporated into a model of a rigid spacecraft rotating in an inertial reference frame.

Let A and B denote a pair of right-handed reference frames, and assume that B has been rotated with respect to A through an angle θ about an axis defined by the unit vector \hat{u} expressed in the basis A. The quaternion of reference frame B with respect to A is defined to be the ordered quadruple

$${}^B q_A = \left\{ \begin{matrix} q_0 \\ \bar{q} \end{matrix} \right\} = \left\{ \begin{matrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{matrix} \right\} = \left\{ \begin{matrix} \cos(\theta/2) \\ \hat{u} \sin(\theta/2) \end{matrix} \right\} \quad (2.1)$$

Quaternions are usually regarded as composed of a scalar component, q_0 , and a vector component, \bar{q} , consisting of the last three elements of the quaternion. Definition (2.1) is more restrictive than typically employed in general discussions. This particular definition is useful when quaternions are used to transform vectors between reference frames.

Quaternion addition and subtraction are the ordinary operations for ordered n-tuples of numbers. Quaternion multiplication is defined by

$$p q = \begin{bmatrix} p_0 q_0 - \bar{p} \cdot \bar{q} \\ p_0 \bar{q} + q_0 \bar{p} + \bar{p} \times \bar{q} \end{bmatrix} \quad (2.2)$$

where \cdot and \times denote the ordinary vector dot and cross products, respectively. The multiplication operation defined by (2.2) has the following interpretation. If ${}^B q_A$ denotes the quaternion of frame B with respect to A, and ${}^C q_B$ denotes the quaternion of C with respect to B, then their product

$${}^C q_A = {}^C q_B {}^B q_A \quad (2.3)$$

is the quaternion that defines the orientation of frame C with respect to frame A. The conjugate of a quaternion q is defined to be

$$q^* = \left\{ \begin{matrix} q_0 \\ -\bar{q} \end{matrix} \right\} \quad (2.4)$$

Definition (2.2) can be used to show that q^* is the multiplicative inverse of q .

Quaternions can be used to transform vectors expressed in one reference frame into another coordinate frame. A vector x^A which is coordinatized in a reference frame A can be transformed into the basis of another reference frame B by the sequence of quaternion products

$$x^B = \begin{pmatrix} B \\ q_A \end{pmatrix}^* x^A q_A^B \quad (2.6)$$

where the vectors x^A and x^B are expressed as quaternions whose scalar parts are zero. It is the simplicity of this expression which motivates the use of (2.1) as the working definition of a quaternion. Note that the operation of transforming a vector requires only arithmetic addition and multiplication, no trigonometric functions are involved.

Throughout this paper, it will be necessary to compute quaternion products of only two types of quaternions: Quaternions associated with rotations of orthogonal reference frames, and quaternions associated with 3-vectors. The scalar part of a quaternion representing a 3-vector is always zero. Since there cannot be more than three independent quantities which characterize rotations in a three-dimensional space, the four elements of a quaternion cannot be independent. A consequence of definition (2.2) is that quaternions which perform transformations between orthogonal reference frames have a Euclidean norm of one; given any three elements of such a quaternion, it is possible to compute the fourth. In particular, the expression

$$q_0 = \sqrt{1 - \sum_{i=1}^3 q_i^2} \quad (2.8)$$

can be used to compute the scalar part of a quaternion from its vector part. From now on, it will be assumed that all quaternions are represented by triples of numbers; hence, the definition of quaternion multiplication (2.2) must be modified to use either zero or (2.8) when the scalar part of the quaternion is needed. With or without subscripts, the symbol q is reserved exclusively for quaternions that perform orthogonal transformations, and quaternions associated with 3-vectors will be denoted by $\begin{Bmatrix} 0 \\ x \end{Bmatrix}$. The notation x will also be used when the interpretation is clear from the context. With these definitions, it is possible to derive a minimal system of first order equations to describe the spacecraft dynamics.

In the following sections, it will be necessary to compute the time rate of change of a quaternion due to the relative rotation of two reference frames. For example, if the quaternion transforms vectors in inertial coordinates into a spacecraft fixed frame, the quaternion will change with time as the spacecraft rotates. The derivative of a quaternion, \dot{q}_B^A , is computed assuming reference frame B is rotating at a relative angular rate ω^B which has been coordinatized in the reference frame B. It can be easily checked that

$$\dot{q}_A^B = \frac{1}{2} \begin{Bmatrix} 0 \\ \omega^B \end{Bmatrix} q_A^B \quad (2.7)$$

In order to derive dynamical equations for a spacecraft equipped with CMGs, it is necessary to have a kinematical model for CMGs. Figure 1. shows a schematic of an Euler mounted double-gimbal CMG. In this figure $\hat{\sigma}_o$ is a unit vector along the outer gimbal axis. This axis is fixed with respect to the spacecraft. The unit vector $\hat{\sigma}_i$ lies along the reference inner gimbal axis corresponding to $\theta_o = 0$, and the third vector in the triad is defined to be orthogonal to $\hat{\sigma}_o$ and $\hat{\sigma}_i$ (i.e. $\hat{h}^0 = \hat{h}(\theta_o=0, \theta_i=0) = \hat{\sigma}_o \times \hat{\sigma}_i$). The CMG momentum vector can now be expressed in the basis $(\hat{\sigma}_o, \hat{\sigma}_i, \hat{h}^0)$ defined relative to a spacecraft fixed frame.

Holding the inner gimbal fixed at zero and rotating the CMG about the outer gimbal axis yields the expression

$$\hat{h}(\theta_o, 0) = \hat{h}^0 \cos \theta_o - \hat{\sigma}_i \sin \theta_o \quad (2.4)$$

for the unit vector along the rotor axis. Rotating the inner gimbal about the rotated inner gimbal axis, $\hat{\sigma}_i$, yields the expression for a unit vector along the rotor axis

$$\hat{h}(\theta_o, \theta_i) = \hat{h}(\theta_o, 0) \cos \theta_i + \hat{\sigma}_o \sin \theta_i \quad (2.5)$$

Substituting (2.4) into (2.5) and scaling by the magnitude of the rotor momentum, h_{mag} , yields

$$h_{dg}(\theta_o, \theta_i) = h_{mag} \left[\left(\hat{h}^0 \cos \theta_o - \hat{\sigma}_i \sin \theta_o \right) \cos \theta_i + \hat{\sigma}_o \sin \theta_i \right] \quad (2.6)$$

as the final expression for the momentum of a double-gimbal CMG.

Equation (2.6) expresses the momentum vector of a CMG in a spacecraft fixed reference frame. In general, the momentum vectors of a group of CMGs will be coordinatized in a common frame. Given a group of CMGs, (2.6) can be used to express the net momentum of the CMG system in a spacecraft-fixed reference frame as a sum over the set of CMGs

$$h_{cmg}(\theta) = \sum_{i=1}^{N_{cmg}} h_{dg}^i(\theta_o, \theta_i) \quad (2.7)$$

where $h_{dg}^i \in \mathbb{R}^3$ is the momentum of CMG i , and $\theta \in \mathbb{R}^m$ denotes the set of gimbal angles.

The attitude of the spacecraft is characterized by a quaternion, q , which defines the relative orientation of a spacecraft-fixed reference frame, denoted B , with respect to an inertial frame. The total angular momentum of the spacecraft in inertial coordinates, h^I , can be transformed into the basis of the spacecraft-fixed frame using (2.6) to yield

$$h^B = q^* h^I q \quad (2.9)$$

The momentum in the spacecraft reference frame, h^B , can be expressed as

$$h^B = h_{cmg}(\theta) + h_{spacecraft} = h_{cmg}(\theta) + I\omega^B \quad (2.10)$$

where ω^B denotes the spacecraft's angular rate, and I denotes the inertia tensor of the vehicle.

Both $h_{cmg}(\theta)$, and ω^B are expressed in the body fixed coordinate frame B . The angular rate of the spacecraft can be determined from the total momentum and the momentum stored in the CMGs by solving (2.10) for ω^B

$$\omega^B = I^{-1} \left(h^B - h_{cmg}(\theta) \right) = I^{-1} \left(q^* h^I q - h_{cmg}(\theta) \right) \quad (2.11)$$

This expression can be substituted into (2.7) to obtain an expression for the derivative of the attitude quaternion in terms of the constant inertia matrix, the constant inertial angular momentum, and the attitude quaternion of the spacecraft.

$$\dot{q} = \frac{1}{2} \begin{pmatrix} 0 \\ I^{-1} \left(q^* h^I q - h_{cmg}(\theta) \right) \end{pmatrix} q \quad (2.12)$$

In order to complete the derivation of the equations for rotation in the absence of external torques, it is necessary to consider the control or input variable. CMG steering algorithms usually choose the set of gimbal rates as the control variable. Since the instantaneous torque produced by a group of CMGs is a function of the gimbal rates only, the gimbal rates are the only choice available when the steering law is required to produce specified torques. The equations that describe the evolution of the gimbal angles are

$$\dot{\theta} = u \quad (2.13)$$

where $\theta, u \in \mathbb{R}^m$. Under the stated assumptions, the dynamics of the combined spacecraft/CMG system are described by the state equations (2.12) and (2.13). These equations describe the attitude dynamics and the evolution of the gimbal angles, respectively.

The gimbal rate equation (2.13) is also useful when it is desirable to include physical limits on the individual gimbal drives. In some circumstances, it may be more realistic to consider torque limits on the gimbal drive motors, especially when double-gimbal CMGs are being used. With double-gimbal CMGs, motion of one gimbal can produce a torque along the axis of the other. The need to back-drive the gimbals can impose restrictions that can not be accounted for without developing a higher-order model. In this discussion, it is assumed that the gimbal rates - as expressed in the individual CMG frames - can be commanded as the control variables. This assumption is consistent with conventional assumptions and does not result from a limitation

imposed by the feedback linearization theory that will be applied to the problem. It may be desirable in future efforts to examine the effects resulting from including the dynamics of the gimbal drive motors in the model.

Feedback Linearization of the Spacecraft Model

The basic idea behind feedback linearization [12,13] is illustrated in Figure 2. Given a nonlinear system of equations of the form

$$\dot{x} = f(x) + \sum_{i=1}^m g_i(x) u_i = f(x) + G(x) u \quad (3.01)$$

the objective is to obtain a nonlinear change of coordinates in the state $F(x)$, a nonlinear feedback $\alpha(x)$, and a linear, invertible change of coordinates in the input $\beta(x)$ such that the input/output behavior of the system illustrated in Figure 2. has a prescribed linear form. Note that this type of linearization is not an approximation, but an exact transformation in which the linear model can be made to have a particularly simple form. Another feature of the transformation shown in the figure is that it is applied externally to the system equations; the transformation does not require internal modifications to a given model (3.01). As a consequence, the feedback transformations employed here are, in principle, implementable in hardware and software.

For the system of interest in this paper, (2.12) and (2.13), the results of [12] can be used to show that there exists a memoryless, nonlinear state transformation $z=F(q,\theta)$, and an input transformation $u=\alpha(q,\theta)+\beta(q,\theta)v$, such that the evolution of the transformed system are described by linear constant coefficient differential equations of the form

$$\dot{z} = A z + B v \quad (3.1)$$

Furthermore, the state and control vectors of the linearized system can be partitioned into the forms $z = (z_1, z_2, z_3)^T$ and $v = (v_1, v_2)^T$, respectively, with $z_1 \in \mathbb{R}^3$, $\dot{z}_1 = z_2$, $\dot{z}_2 = v_1$ and $\dot{z}_3 = v_2 \in \mathbb{R}^{m-3}$. These definitions imply that the matrices (A,B) must have the special block structured forms

$$A = \begin{bmatrix} 0 & I_3 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 & 0 \\ I_3 & 0 \\ 0 & I_{m-3} \end{bmatrix} \quad (3.2)$$

This assumption regarding the structure of the pair (A,B) can be made without loss of generality, since all achievable linear behaviors can be obtained by applying linear feedback, linear input transformations, and linear state transformations to the system defined by (3.1) and (3.2) [17]. In other words, the forms (3.2) were chosen to simplify the problem of obtaining the linearizing

transformations; once the system is made to behave like (3.2), a separate design procedure will yield a control system that causes (3.2) to have desired characteristics.

A simple justification for the form of the matrices (3.2) is as follows. The linear system defined by (3.1) and (3.2) contains three sets of independent double integrators. These second-order subsystems are related to the second-order rotational dynamics of the spacecraft, and the input variable v_1 is related to the spacecraft acceleration. Since the state components z_3 evolve independently of the spacecraft attitude, trajectories of the variables z_3 can be interpreted as *null-motions* of the actuators. In other words, these states are associated with the redundant degrees of freedom in the CMG system.

Once computed, the feedback transformation simplifies the original system equations in two important ways. First, the nonlinear spacecraft equations are transformed to a very simple linear form; familiar techniques from linear systems theory can then be applied to derive control laws for the spacecraft. Second, the linearized system of equations contains state variables that are explicitly related to the available freedom in the choice of gimbal trajectories. In the sequel, it will be shown how to construct the state and input transformations from a particular choice of z_1 and z_3 for which the transformations are known to exist.

Even though quaternions have been chosen to represent attitude in the model equations, any set of independent attitude variables can be chosen for z_1 . Once the variables z_1 are chosen, it is necessary to obtain the functional relationship

$$z_1 = F_1(q) \quad (3.4)$$

relating the chosen variables to the attitude quaternion of the model. This expression can be differentiated to obtain the value of z_2 in terms of the state variables (q, \dot{q})

$$z_2 = \dot{z}_1 = [D_q F_1(q)] \dot{q} = F_2(q, \dot{q}) \quad (3.5)$$

where (2.12) is substituted to eliminate \dot{q} . The notation $[D_q F(q)]$ has been introduced to denote the derivative of a vector valued function, F , with respect to a vector variable, q . The i^{th} column of the derivative matrix consists of the partial derivatives of the elements of $F(q)$ with respect to the scalar variable q_i .

Although a more general choice is possible, z_3 will be chosen as follows. Assuming that the CMG configuration is not singular, it is possible to order the gimbal angles of the vector θ such that the rank of the 3×3 matrix

$$A_1(x) = \begin{bmatrix} D_{\theta_1} h_{\text{cmg}}(\theta), D_{\theta_2} h_{\text{cmg}}(\theta), D_{\theta_3} h_{\text{cmg}}(\theta) \end{bmatrix} = D_{\phi} h_{\text{cmg}}(\theta) \quad (3.6)$$

is three. Here the symbol ϕ has been used to denote the vector consisting of the first three elements of the ordered gimbal vector. Nonsingularity of this matrix implies that the torques produced by the three gimbal rates $\{u_1, u_2, u_3\}$ span three-space. The component of the linearized state variable

represented by z_3 is defined to be

$$z_3 = \begin{Bmatrix} \theta_4 \\ \vdots \\ \theta_m \end{Bmatrix} \quad (3.7)$$

This completes construction of the coordinate transformation that maps the state variables of the model, $x=(q,\theta)$, to those of the linearized system.

It now remains to construct the feedback and input transformation. By assumption, the control variable v has the interpretation

$$v = \begin{Bmatrix} v_1 \\ v_2 \end{Bmatrix} = \begin{Bmatrix} \ddot{z}_1 \\ \dot{z}_3 \end{Bmatrix} = \frac{d}{dt} \begin{Bmatrix} z_2 \\ z_3 \end{Bmatrix} \quad (3.3)$$

This expression can be evaluated by substituting (3.5) and (3.7), and then differentiating to yield

$$v = \begin{bmatrix} [D_q F_2] & [D_\theta F_2] \\ 0 & 0 \quad I_{m-3} \end{bmatrix} \begin{Bmatrix} \dot{q} \\ \dot{\theta} \end{Bmatrix} = \begin{bmatrix} [D_q F_2] \\ 0 \end{bmatrix} \dot{q} + \begin{bmatrix} [D_\phi F_2] [D_z F_2] \\ 0 \quad I_{m-3} \end{bmatrix} \dot{\theta} \quad (3.9)$$

Once again noting that the quaternion derivative is a function of (q,θ) and using the fact that $\dot{\theta} = u$, the previous equation can be written in the form

$$v = w(q,\theta) + A(q,\theta) u \quad (3.10)$$

Note that if the matrix $A(q,\theta)$ is invertible, then (3.10) can be solved to yield an input transformation of the required form

$$u = [A(q,\theta)]^{-1} w(q,\theta) + [A(q,\theta)]^{-1} v = \alpha(q,\theta) + \beta(q,\theta) v \quad (3.11)$$

or equivalently,

$$u = \begin{bmatrix} \alpha_1 \\ 0 \end{bmatrix} + [\beta_1 \beta_2] \begin{Bmatrix} v_1 \\ v_2 \end{Bmatrix} \quad (3.110)$$

Necessary and sufficient conditions for (3.10) to be solvable for u are that the function F_1 in (3.4) be invertible, and the matrix A_1 in (3.6) be nonsingular. From (3.9), it is clear that the matrix $A(q,\theta)$ will be invertible if the 3×3 submatrix in the upper left corner is nonsingular. This matrix is given by

$$\begin{aligned}
D_\phi F_2(q, \theta) &= [D_q F_1(q)] [D_\phi \dot{q}] \\
&= -\frac{1}{2} [D_q F_1(q)] \left[D_\phi \left(\begin{Bmatrix} 0 \\ h_{cmg} \end{Bmatrix} q \right) \right]
\end{aligned} \tag{3.12}$$

By the implicit function theorem, invertibility of F_1 implies nonsingularity of the matrix $D_q F_1(q)$; therefore, it only remains to show that the remaining 3x3 matrix is nonsingular. Observing that

$$D_\phi \left[\begin{Bmatrix} 0 \\ h_{cmg} \end{Bmatrix} q \right] = D_\phi \left[\begin{bmatrix} h_{cmg}^1 & 0 & -h_{cmg}^3 & h_{cmg}^2 \\ h_{cmg}^2 & h_{cmg}^3 & 0 & -h_{cmg}^1 \\ h_{cmg}^3 & -h_{cmg}^2 & h_{cmg}^1 & 0 \end{bmatrix} \begin{Bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{Bmatrix} \right] \tag{3.13}$$

it can be seen that the rows of the above matrix consist of linear combinations of the rows of the matrix A_1 in (3.6). Because (3.13) implies that one of the coefficients in each linear combination must differ in sign from the corresponding coefficient in the other two combinations, each row of the matrix (3.13) must be a unique combination of the rows of A_1 . The converse follows immediately: Singularity of $D_\phi \dot{q}$ implies singularity of (3.6), and singularity of $D_q F_1(q)$ implies that $F_1(q)$ is not invertible.

In concluding this section, it is useful to discuss the implications when one of the necessary and sufficient conditions fails. Singularity of $D_\phi \dot{q}$ implies that the CMG configuration is singular, since singularity of A_1 implies that there is a direction in which the CMG system can not instantaneously project momentum. A later section presents a method for dealing with this eventuality. Noninvertibility of $F_1(q)$ implies that it is not possible to associate a unique physical orientation of the body with the parameters z_1 . For example, the quaternion associated with a 180 degree rotation is the same regardless of the axis of rotation. This difficulty arises due to the choice of attitude variables, and is not a limitation of the methodology developed here; any attitude control law employing these parameters will have to deal with these singularities. In practice, this is not a limitation since the singularity can be eliminated by merely changing the reference attitude for which the attitude variables are defined.

Gimbal Rate Distribution by Normed Approximation

In this section, an approach to distributing control effort among a group of CMGs is developed that employs the linearizing transformation developed previously. Although the system to be controlled has been made linear by the transformation (3.11), the steering problem has not been solved. Effectively, the linearizing transformation decouples the problems of spacecraft

control and redundancy management; the problem of distributing control effort among the available actuators has not been addressed. If it were not for the kinematic redundancy in the CMG system, the solution to the linearization problem would be adequate and unique; no steering problem would exist.

A natural choice of gimbal rates would be one that requires a minimal effort to produce a desired influence on the spacecraft. This would be adequate if there were no dynamics associated with the CMGs. In the problem at hand, there are certain configurations or CMG states that are to be avoided. Consequently, there may be a desired value u_d for the gimbal rates. The problem of computing gimbal rates, u , that are close to the desired value can be stated as a normed approximation problem

$$\min_u \|u_d - u\| \quad (4.1)$$

subject to the constraint that a specified acceleration be produced. Note that the feedback part of the linearizing transformation (3.11) yields an affine parameterization of all gimbal rates that produce a specified attitude acceleration. Once the first three elements of the control vector of the linear system have been fixed, the remaining elements can be selected arbitrarily without changing the commanded acceleration. The parameterization effectively eliminates the torque constraint associated with the steering problem, and reduces the number of variables that need to be considered in by three. Assuming that the component v_1 of the linearized control vector is specified and can be regarded as fixed, substitution of (3.110) into (4.1) yields

$$\min_{v_2} \|a - \beta_2 v_2\| \quad (4.2)$$

where $a = u_d - \alpha - \beta_1 v_1$. From a practical point of view, it is useful to impose limits on the individual inputs of the form

$$-u_p^i \leq u_i \leq u_p^i \quad (4.3)$$

where u_p^i is a physical limitation associated with gimbal i . Fast numerical methods exist for solving problem (4.2) with constraints (4.3) in three important norms: The 1-norm defined by

$$\|x\|_1 = \sum_{i=1}^n |x_i| \quad (4.4)$$

the 2-norm is the usual Euclidean norm defined by

$$\|x\|_2 = \left\{ \sum_{i=1}^n |x_i|^2 \right\}^{\frac{1}{2}} \quad (4.5)$$

and the ∞ -norm is defined to be

$$\|x\|_\infty = \max_{x_i} |x_i| \quad (4.6)$$

The constrained problem in the 2-norm can be formulated directly as a quadratic programming problem [18]. The 1- and ∞ -norm problems can be formulated as linear programs as shown in [19] and [20], respectively.

An Algorithm for Computing Desired Rates for Double-Gimbal CMGs

In the following, an algorithm for computing a desired set of gimbal rates for a group of double-gimballed CMGs is developed. The algorithm is based on computing a desired rotation vector for each CMG. The direction and magnitude of the rotation vector of a CMG rotor correspond to the axis and rate of rotation, respectively, of the rotor due to gimbal motion. The rotation vectors are chosen to move the CMG rotors to maximize the amount of momentum which can be projected instantaneously in any direction, to avoid line-up with the commanded torque vector, and to keep inner-gimbal angles small. The desired rotation vector for a CMG is computed as the sum of a set of rotation vectors each of which individually attempts to achieve one of these objectives. From the desired rotation vector, desired gimbal rates are derived for each CMG. This section concludes with a brief explanation of how the proposed steering law is incorporated into an attitude control system.

One method for maintaining good three-axis controllability with double-gimballed CMGs is to keep the angles between rotors as large as possible. Achieving this goal corresponds to maximizing the minimum torque that can be projected in any direction and corresponds to keeping three-axis controllability of the spacecraft high in all directions. Since singular states of double-gimballed CMG systems are always associated with a mutual alignment of rotors, this is also equivalent to avoiding singularities. For each CMG, the term which achieves this in the rotation vector computation is

$$\sigma_1^i = \sum_{i \neq j} \frac{u_p}{\pi} \left[\cos^{-1}(\hat{h}^i \cdot \hat{h}^j) - \pi \right] \text{unit}(\hat{h}^i \times \hat{h}^j) \quad (4.7)$$

where u_p denotes the peak gimbal rate, \hat{h}^i denotes a unit vector along rotor axis i , and the cross product is normalized to have unit norm. Rotating each CMG in this way will move the rotors to a configuration in which they are distributed symmetrically about the total CMG momentum vector.

If a rotor is allowed to align with the commanded torque, then that rotor cannot contribute to the torque. The term which steers rotors away from such configurations is

$$\sigma_2^i = \left\{ \begin{array}{ll} 0 & \text{if } \cos^{-1}(\hat{h}^i \cdot \hat{\tau}) \geq k^\tau \\ \sigma_1^i = \sum_{i \neq j} \frac{u_p}{k^\tau} \left[\cos^{-1}(\hat{h}^i \cdot \hat{\tau}) - k^\tau \right] \text{unit}(\hat{h}^i \times \hat{\tau}) & \text{if } \cos^{-1}(\hat{h}^i \cdot \hat{\tau}) < k^\tau \end{array} \right\} \quad (4.8)$$

where $\hat{\tau}$ denotes a unit vector in the direction of the requested torque, k^τ is a constant, and the cross product is normalized as before. To avoid numerical difficulties, an arbitrary gimbal axis is used in the neighborhood of an exact line-up of vectors in (4.7) and (4.8).

To simplify notation in defining the next two contributors to the rotation vector, the function Φ is defined by

$$\Phi(a,b,c,d) = \left\{ \begin{array}{ll} 0 & \text{if } \left| \frac{a}{b} \right| \leq c \\ \frac{a}{|a|} \frac{|a/b| - c}{d - c} & \text{if } c < \left| \frac{a}{b} \right| \leq d \\ 1 & \text{if } \left| \frac{a}{b} \right| > d \end{array} \right\} \quad (4.9)$$

where (a,b,c,d) are constants with $|a| \leq b$, and $0 \leq c \leq d \leq b$.

It is desirable to keep the inner gimbal angles small, both to avoid potential hardware stops, and because the torque due to the outer gimbal rate is proportional to the cosine of the inner gimbal angle. A rotation vector which reduces the inner gimbal displacements is

$$\sigma_3^i = -u_p \Phi(\theta_i, \theta_{\text{stop}}, k_1^{\text{inner}}, k_2^{\text{inner}}) \hat{\sigma}_{\text{inner}} \quad (4.10)$$

where θ_i denotes the displacement of gimbal i , θ_{stop} is the displacement to a gimbal stop, k_1^{inner} , k_2^{inner} are constants, and $\hat{\sigma}_{\text{inner}}$ is a unit vector along the inner gimbal axis.

In studying several maneuvers, it was found necessary to include a term in the calculation

of the desired rotation vector which would rotate the outer gimbal angle if the inner gimbal was being forced against a stop. Associated with each CMG is the rotation rate

$$\sigma_4^i = \pm u_p \Phi(\theta_i, \theta_{stop}, k_1^{outer}, k_2^{outer}) \hat{\sigma}_{outer} \quad (4.11)$$

where θ_i and θ_{stop} are as before, k_1^{outer} , k_2^{outer} are constants, and $\hat{\sigma}_{outer}$ is a unit vector along the outer gimbal axis. The initial sign of this rotation is chosen to move the rotor away from the inner gimbal rotation plane, and is not changed until σ_4^i is set to zero. The need for such a term will be explained in connection with one of the examples.

For each CMG, the desired rotation vector is found by taking a weighted combination of the four rotation vectors defined above.

$$\sigma_d^i = \sum_{j=1}^4 k_j \sigma_j^i \quad (4.12)$$

From this composite rotation vector, it is possible to compute a unique set of gimbal rate commands that produce the desired rotor motion for the given CMG by solving the linear equation

$$\sigma_d^i \times \hat{h}^i = u_d^{inner} (\hat{\sigma}_{inner} \times \hat{h}^i) + u_d^{outer} (\hat{\sigma}_{outer} \times \hat{h}^i) \quad (4.13)$$

for the desired inner and outer gimbal rates. Proceeding in this way for each CMG, a set of desired gimbal rates, u_d , is computed, and the normed approximation strategies of the previous section can be used to calculate actual gimbal rates based on these desired rates. In the following subsection, an implementation of a steering law based on this desired rate computation and the approximation algorithms is described.

Implementation of the Proposed Algorithm

In this section, an algorithm is described which implements a CMG steering law based on normed approximation and the desired rate computation of the previous section. The program has been implemented in FORTRAN on a VAXTM computer along with a rigid body simulation of a spacecraft. A flow chart of the steering algorithm is shown in Figure 3. The inputs are the current CMG gimbal angles, the spacecraft attitude quaternion, the derivative of the quaternion, and the second derivative of the quaternion, which is determined by an autopilot or control law. Since quaternions are algebraically related to any set of attitude variables, the proposed implementation is quite general.

The steering algorithm first tests the CMG gimbal angles to see if the CMG configuration is near a singular state. If the gain function is above a certain threshold, a routine that monitors the transformation is called. The routine MONITOR checks that the three gimbal rates that do not

appear in the linearized control vector are capable of producing torques in all directions. Their ability to control the spacecraft is determined by the determinant of the matrix formed by the three columns of the matrix (3.6). If the determinant is below a certain threshold, a new combination of three gimbal angles is found which produces the largest determinant, and these three gimbal rates will not appear in the transformed control. Note that this effectively amounts to reindexing the gimbal rates. These changes occur internally to the transformation illustrated in Figure 2, and are transparent to both the attitude control functions and the normed approximation problem.

If the CMG rotors are near a singular configuration, then there will be no combination of three CMGs that can produce torques adequately in all directions. Most steering algorithms do not consider this situation at all; even though no steering algorithm can really guarantee absolute singularity avoidance. The most common symptom of such a failure is termed gimbal lock, which occurs when the CMGs become stuck in a singular configuration due to numerical errors or because the steering algorithm is attempting to normalize gimbal rates to maintain their boundedness. It is necessary that a steering algorithm continue to operate in a sensible manner under these circumstances.

Near singular states, the linearization problem is not solvable as stated. The numerical difficulty that arises in computing the transformations is due to the fact that the matrix $A(x)$ in (3.10) is nearly singular. This matrix has the form

$$A(x) = \begin{bmatrix} A_1(x) & A_2(x) \\ 0 & I_{m-3} \end{bmatrix} \quad (4.14)$$

The MONITOR routine attempts to order the gimbal angles to guarantee numerical invertibility of $A_1(x)$; however, near a singular state, no such ordering exists. The matrix A_1 can be decomposed into

$$A_1 = V \Lambda V^T \quad (4.15)$$

where $\Lambda = \text{diag}[\lambda_1, \lambda_2, \lambda_3]$ is the diagonal matrix of eigenvalues of A_1 , and $V = \text{col}[e_1, e_2, e_3]$ is the matrix whose columns are the orthonormalized eigenvectors. The problem of computing the linearizing feedback, (3.10) can be transformed to the form

$$\begin{bmatrix} V^T & 0 \\ 0 & I_{m-3} \end{bmatrix} v = \begin{bmatrix} V^T \dot{q} \\ \dot{z}_3 \end{bmatrix} = \begin{bmatrix} \Lambda V^T & V^T A_2 \\ 0 & I_{m-3} \end{bmatrix} \{ w(q, \theta) + u \} \quad (4.16)$$

where the third row in the matrix on the right is nearly zero - assuming the eigenvalues decrease along the diagonal of Λ . Eliminating this row from the equation and increasing the dimension of the identity matrix by one yields the feedback transformation

$$u = \alpha(q, \theta) + \begin{bmatrix} \begin{bmatrix} -\lambda_1 e_1 \\ -\lambda_2 e_2 \end{bmatrix} & \begin{bmatrix} -e_1 \\ -e_2 \end{bmatrix} A_2 \\ 0 & I_{m-2} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} -e_1 \\ -e_2 \end{bmatrix} \ddot{q} \right. \\ \left. \dot{z}_3 \right\} \quad (4.17)$$

Near a singular state, there is a direction about which the CMGs can produce no acceleration. This modification to the algorithm projects the desired acceleration onto the space orthogonal to the singular direction, and control of the spacecraft about the singular direction is replaced by an extra gimbal rate in the transformation. The spacecraft will not be controlled in the singular direction; however, control will be maintained along the other axes. Controllability will be recovered in the singular direction after the singular orientation has been eliminated. Depending on the torque and the orientation of the singular direction, singularity may be a problem for only a very short time.

After the program resolves potential computational difficulties, the linearizing state and feedback transformations are evaluated in the routine TRANSFORMS. The routine CMG_DISTRIB is then called to compute the desired set of gimbal rates based on the current state of the CMG system. The routine APPROX calls the appropriate subroutines for formulating and solving the desired normed approximation problem. The 1-norm and 2-norm approximation problems are solved using the subroutines DLPRS and QPROG, respectively, from release 10.0 of the IMSL™ subroutine library. The ∞ -norm approximation problem is solved using subroutine ZX3LP from release 9.2 of the IMSL™ library. The output of the steering program is a set of gimbal rates, u , which produce the commanded \ddot{q} when possible.

This steering algorithm has been implemented as part of a simple simulation of a rigid spacecraft. The equations of motion are integrated using a fourth-order Runge-Kutta integration scheme. Before computing an integration step, a subroutine is called that can implement a variety of attitude control functions, and then the CMG steering algorithm is called. The chosen gimbal rates are then passed to a subroutine that executes a single integration step. The following section presents results from simulations using the program described in this subsection.

Simulation Examples

In this section, several simulations are presented to demonstrate the behavior of the proposed steering algorithm. The first example demonstrates the need for the fourth term in the rotation axis computation, (4.11). This is an example of a situation in which the steering algorithm must deal explicitly with a problem that conventional gradient based methods would fail to

recognize. The next example shows how the algorithm redistributes the CMG rotors when they are initialized in a singular configuration.

The inertia properties of the vehicle are listed in Table 1. and represent the rephased dual-keel space station, after assembly sequence flight 7 [21]. Attitude control torques are provided by four parallel-mounted double-gimballed CMGs. In the parallel mount configuration, the outer gimbals are aligned with the vehicle pitch axis and the inner gimbals are aligned along roll. Each CMG rotor possesses 5000 ft-lb-sec of angular momentum and can be gimballed about inner and outer mounts at peak rates of 5 deg/sec. Inner gimbal displacements are limited to less than 90 deg and outer gimbals may rotate continuously. Figure 4. shows a schematic diagram of the vehicle along with the body reference frame, and the CMG mounting configuration is shown in Figure 5. The set of constants used by the steering law to compute the desired gimbal rotation vectors are shown in Table 2. Although these simulations employ parallel CMG mounting, the structure of the steering law by no means requires this, and is amenable to any mounting configuration.

In this example, the CMG momentum vectors are initialized so that rotors 1 and 4 align with the positive and negative yaw axes, respectively, and rotors 2 and 3 align with the positive and negative roll axes, respectively. A constant torque of -100 ft-lb is commanded along the negative roll axis of the vehicle. This torque will cause the CMG rotors to align along the positive roll axis. Because of the particular symmetry of the situation, the desired rotation vector of rotor 2 consistently lies along its inner gimbal axis. As a consequence, this gimbal will eventually be rotated against its stop. The outer gimbal rotation term (4.11) was added to the rotation vector computation to prevent this from happening.

Figure 6. and Figure 7. show the gimbal displacements and gimbal rates from the simulation. At about 125 sec. into the simulation, the outer gimbal of rotor 2 begins to rotate at its peak rate. Initially, the inner gimbal axis is aligned with the negative yaw axis of the vehicle. As the outer gimbal angle rotates, the inner gimbal axis eventually has a component along the positive yaw axis and the gimbal rate must change sign in order to continue projecting more momentum in the positive roll direction. This causes the sudden change of sign in the inner gimbal rate. Figure 8. shows that rotating the outer gimbal causes a drop in the gain function shortly before $t=125$ secs. Steering laws based on gain or controllability optimization would never choose to rotate the outer gimbal in this situation, and the inner gimbal would be forced against a stop. A similar statement applies to steering laws that attempt to avoid rotor line-up. The approach [] which keeps all inner gimbal angles equal, will not move rotor 3 from its initial position. Note that the approximation algorithm is bounding the rate of outer gimbal 3 at its peak value.

In the first example, an angular rate change is commanded with the CMGs beginning in an initially singular configuration. The attitude control law commands an angular acceleration

proportional to a rate error in the pitch direction. The rotors are initialized in a zero momentum state with rotors 1 and 2 aligned with the positive roll axis, and rotors 3 and 4 nearly aligned with the negative roll axis; this configuration is numerically singular. The gimbal angle responses are shown in Figure 9. The inner gimbals must project momentum along the positive pitch axis to achieve the rate change, but the outer gimbals move until they are evenly distributed about the net CMG momentum vector in order to alleviate the singular condition and attain a best orientation. In Figure 10, the gain function is shown. The gain is initially almost zero and increases rapidly to about 80% of its maximum value (recall that gain is not being optimized). It then decreases as the CMGs must project momentum in the pitch direction to maintain the rate change. This example illustrates that the proposed algorithm functions in the vicinity of a singular state. Even if the requested torque had been along the singular direction, null motions would drive the rotors to a configuration which could produce the desired torque.

Conclusions

This paper has presented an approach to obtaining gimbal rates for the control of CMG equipped spacecraft. A feedback linearizing transformation was used to linearize the spacecraft dynamics, and then the transformation's special form was exploited to obtain a normed approximation problem that controls the spacecraft in a commanded way while redistributing the gimbals in a desired direction. An algorithm for computing a desired set of gimbal rates was proposed for systems of double-gimbal CMGs, and demonstrated with two examples in which existing algorithms are known to fail.

Using the proposed approach, the problem of developing a steering law involves designing an algorithm for the computation of desired gimbal rates. This is conceptually simpler than having to determine a function whose gradient produces desired rates. A difficulty encountered with algorithms that seek local optima of functions is that the algorithms can get stuck in a locally optimal configuration. For the parallel mount configuration of four CMGs, the technique demonstrated here has a unique, stable, optimally controllable configuration for the rotors provided that the CMGs project momentum in either the pitch or the yaw direction (the placement of the rotors is unique, not the location of the rotors by index). If the CMGs project momentum only in the pitch direction, any optimally controllable configuration can be rotated as a whole about the outer gimbal axis and still be optimal.

Most steering algorithms that have been proposed use the pseudoinverse of the matrix Dh_{cmg} to determine a torque producing set of gimbal rates. Because of the special form of the matrix $A(x)$ in (3.10), the computation of the linearizing transformation requires considerably fewer arithmetic operations. In addition, because it is never necessary to invert more than the 3×3

submatrix A_1 , the numerical complexity of the method described in this paper does not increase with the number of CMGs. Because the linearizing transformation developed here satisfies a torque constraint implicitly, the feedback transformation can be incorporated into an existing steering algorithm to reduce the computations required, and to reduce the number of variables considered by three.

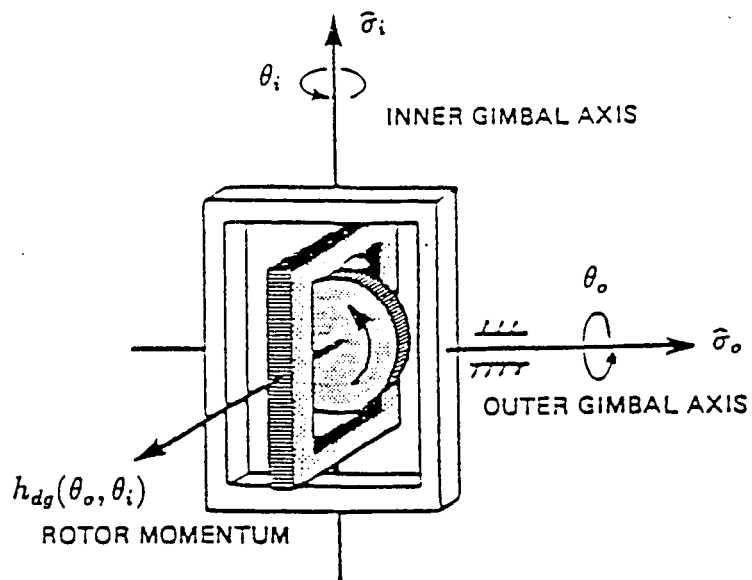
Although the technique proposed here is generally applicable to steering single-gimbal CMGs, the problem of determining desired rates is considerably more complicated. It may be necessary to relax constraints on the spacecraft control to insure adequate singularity avoidance capability. This problem should be a subject for continuing research.

References

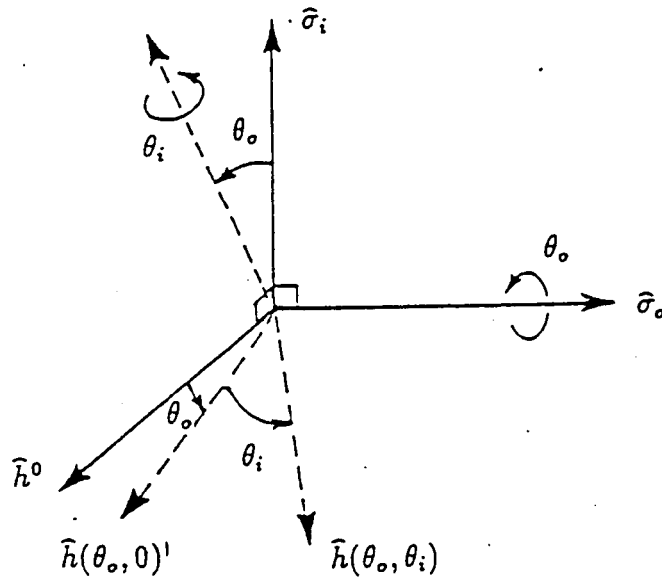
- [1] Bedrossian, N., Steering Law Design for Redundant Single Gimbal Control Moment Gyro Systems, S.M. Thesis, Massachusetts Institute of Technology, 1987.
- [2] Yoshikawa, T., "A Steering Law for Three Double-Gimbal Control Moment Gyro Systems," NASA, TM - 82390, Jan. 1981.
- [3] Paradiso, J., "A Highly Adaptable Steering/Selection Procedure for Combined CMG/RCS Spacecraft Control," *10th Annual {AAS} Guidance and Control Conference*, Jan. 1987.
- [4] Kurokawa, K., Yajima, N., and Usui, S., "A new steering law of a single-gimbal cmg system of pyramid configuration," *Xth IFAC Symposium on Automatic Control in Space*.
- [5] Cornick, D., "Singularity Avoidance Control Laws for Single Gimbal Control Moment Gyros," *AIAA Guidance and Control Conference*, Aug. 1979.
- [6] Kennel, H., "Steering Law for Parallel Mounted Double-Gimballed Control Moment Gyros - Revision A", NASA TM - 82390, Jan. 1981.
- [7] Nakamura, Y. and Hanafusa, H., "Inverse Kinematic Solutions with Singularity Robustness for Robot Manipulator Control," *ASME J. Dyn. Sys., Meas., and Cont.*, Vol. 108, 1986, pp. 163-171.
- [8] Lenorovitz L., "Soviets Use New Gyros to Stabilize MIR Station", *Aviation Week and Space Technology*, Nov. 1987.
- [9] Kennel, H., "A Control Law for Double-Gimballed Control Moment Gyros Used for Space Vehicle Attitude Control", NASA, 1970.
- [10] Architectural Control Document: Guidance, Navigation, and Control System, Space Station Program Office, NASA ACD 3/JSC 30259, Jan. 1987.
- [11] Ref. [3]
- [12] Hunt, L., Su, R., and Meyer, G., "Global transformations of nonlinear systems", *IEEE Trans. on Auto. Control*, Vol. 28, 1983, pp. 24-31.

- [13] Jakubczyk, B. and Respondek, W., "On Linearization of Control Systems," Bull. Acad. Polon. Sci., Ser. Sci. Math. Astronom. Phys., Vol. 28, 1980, pp. 517-522.
- [14] Dzielski, J., "A Feedback Linearization Approach to Spacecraft Control Using Momentum Exchange Devices", Ph.D. Thesis, Massachusetts Institute of Technology, 1988.
- [15] Dwyer T., "Exact Nonlinear Control of Large Angle Rotational Maneuvers", *IEEE Trans. on Auto. Control*, Vol. 29, 1984, pp. 769-774.
- [16] Dwyer T., "Exact Nonlinear Control of Spacecraft Slewing Maneuvers with Internal Momentum Transfer", *AIAA J. Guidance*, Vol. 9, 1986, pp. 240-247.
- [17] Brunovsky, P., "A Classification of Linear Controllable Systems", *Kybernetika*, Vol. 6, 1970, pp. 173-188.
- [18] Wismer, D. and Chattergy, R., *Introduction to Nonlinear Optimization: A Problem Solving Approach*, North-Holland, New York, 1978.
- [19] Fisher, W., "A Note on Curve Fitting with Minimum Deviations by Linear Programming", *J. Amer. Stat. Assoc.*, June 1961, pp. 359-362.
- [20] Kelley, J., "An Application of Linear Programming to Curve Fitting", *J. Soc. Indust. Appl. Math.*, Vol 6, 1958, pp. 15-22.
- [21] Macloed, T., Rigsby, K., and Friedman, M., "Rephased Dual Keel Mass Properties and TEA Analysis", LEMSCO, Aug. 1986.

Figure 1. Schematic Drawing and Reference Frame for a Double-Gimbal CMG.

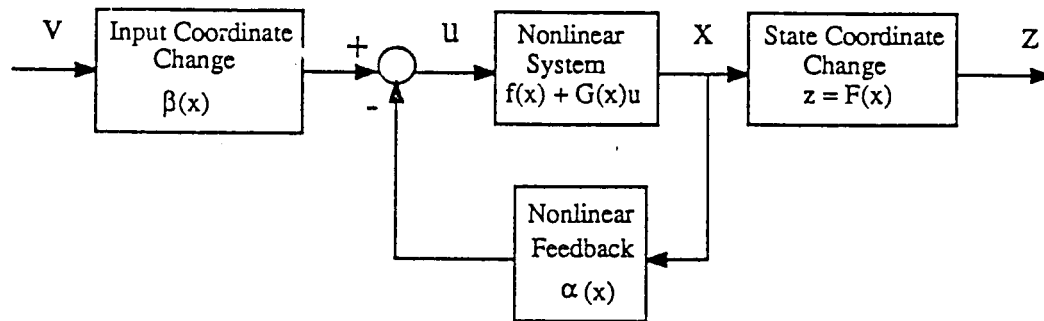


a.) Schematic.



b.) Reference frame.

Figure 2. Implementation of a Linearizing Feedback Transformation.



$$\begin{array}{ccc}
 & u = \alpha(x) + \beta(x) v & \\
 \dot{x} = f(x) + G(x) u & \longleftrightarrow & \dot{z} = A z + B v \\
 & z = F(x) &
 \end{array}$$

Figure 3. Flow Chart of Attitude Control Algorithm.

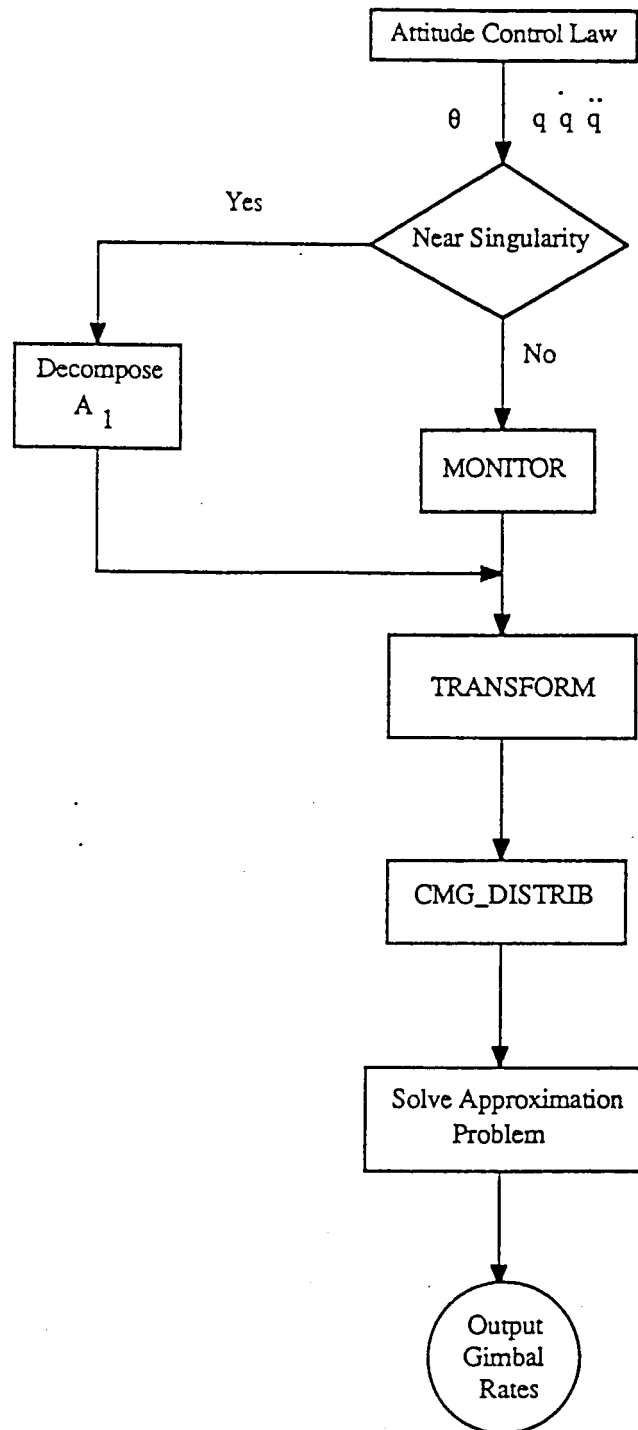


Table 1. Inertia Properties for Re-phased Dual-Keel Space Station.

Inertia Matrix <i>slug-ft</i> ²		
2.425×10^7	-1.917×10^5	3.060×10^5
-1.917×10^5	3.556×10^6	2.333×10^5
3.060×10^5	2.333×10^5	2.515×10^7

Figure 4. Schematic of Rephased Dual-Keel Space Station; Flight 7.

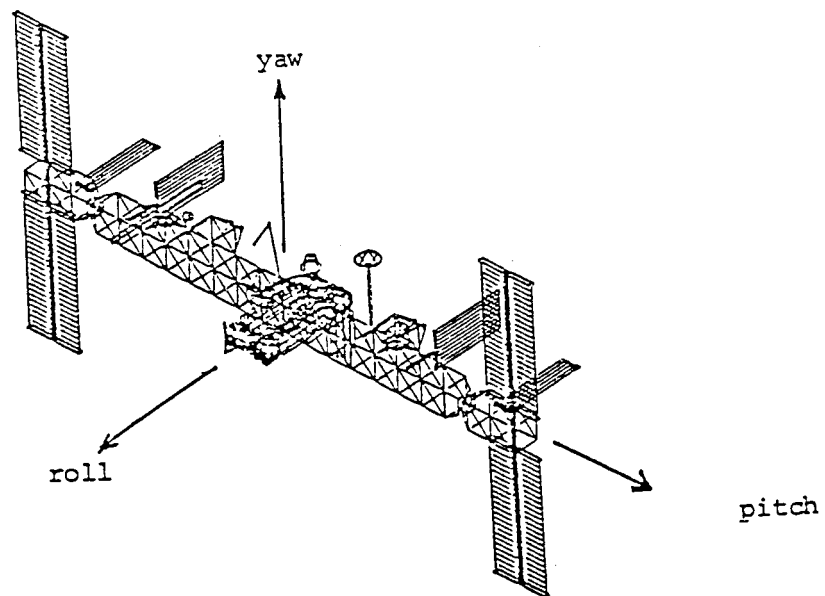


Figure 5. Reference Configuration for Four Parallel Mounted CMGs.

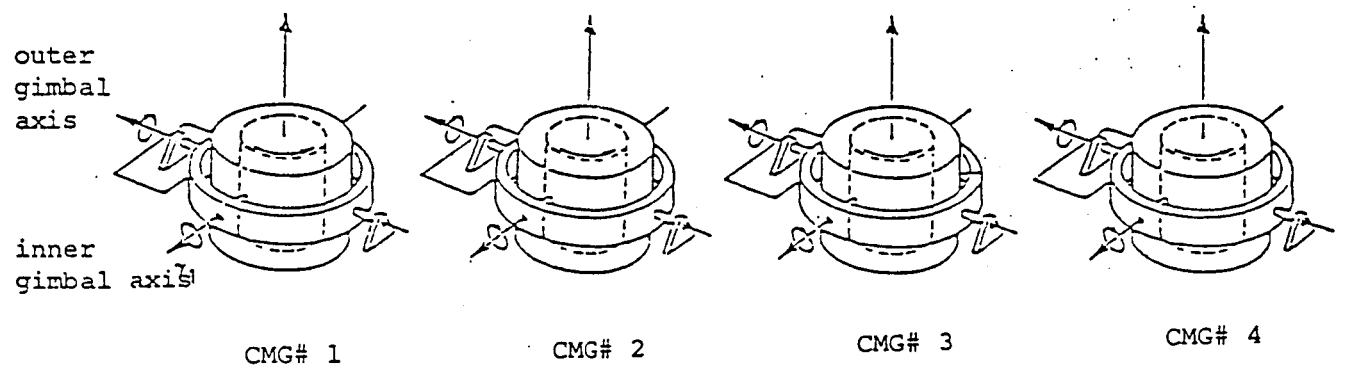


Table 2. Constants Used in Rotation Vector Computation.

Relative Weights	Rotation Vector Constants
$k_1 = 0.2165$	
$k_2 = 1.0$	$k_r = 1.57$
$k_3 = 1.0$	$k_1^{inner} = 0.0$ $k_2^{inner} = 0.6$
$k_4 = 1.0$	$k_1^{outer} = 0.6$ $k_2^{outer} = 0.7$

Figure 6. Gimbal Angles; Momentum Saturation along Roll, 2-norm (674).

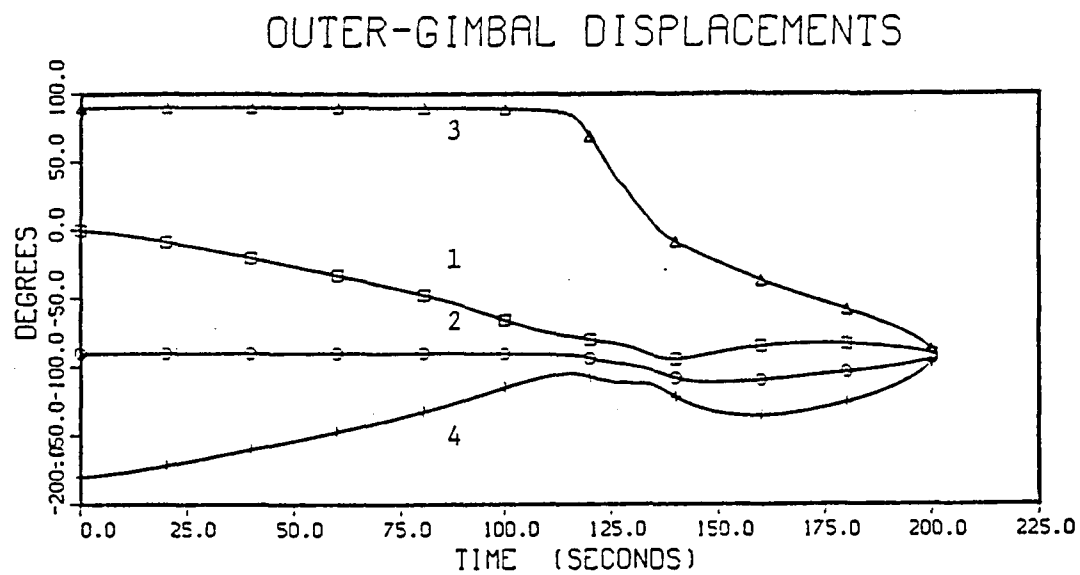
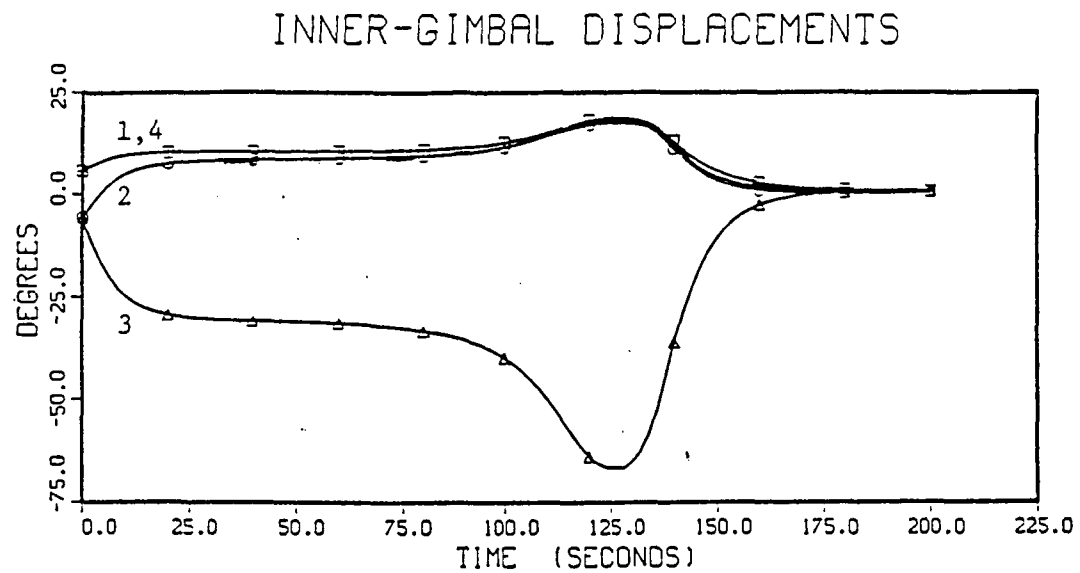


Figure 7. Gimbal Rates; Momentum Saturation along Roll, 2-norm (674).

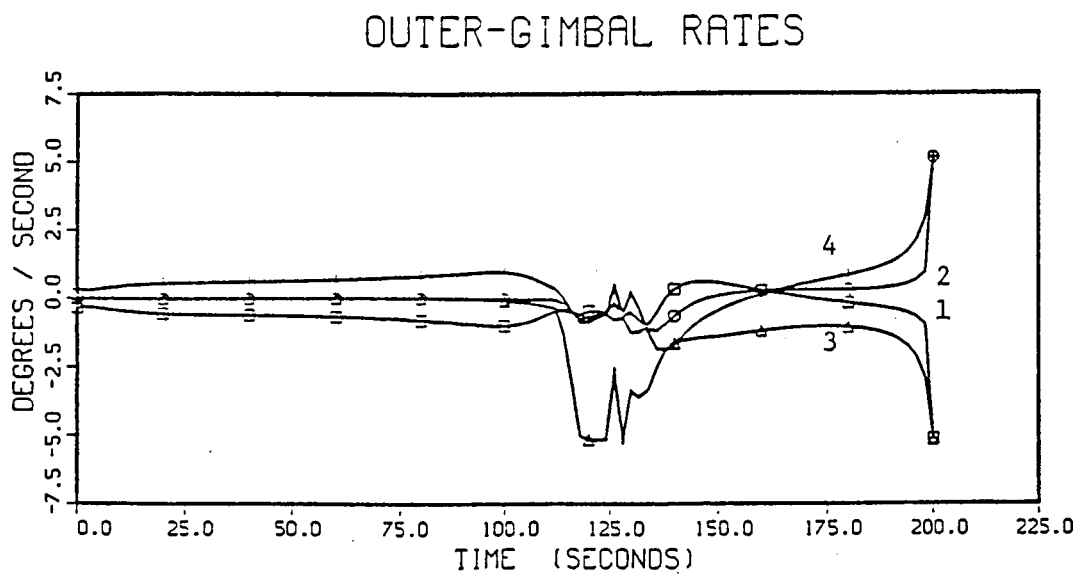
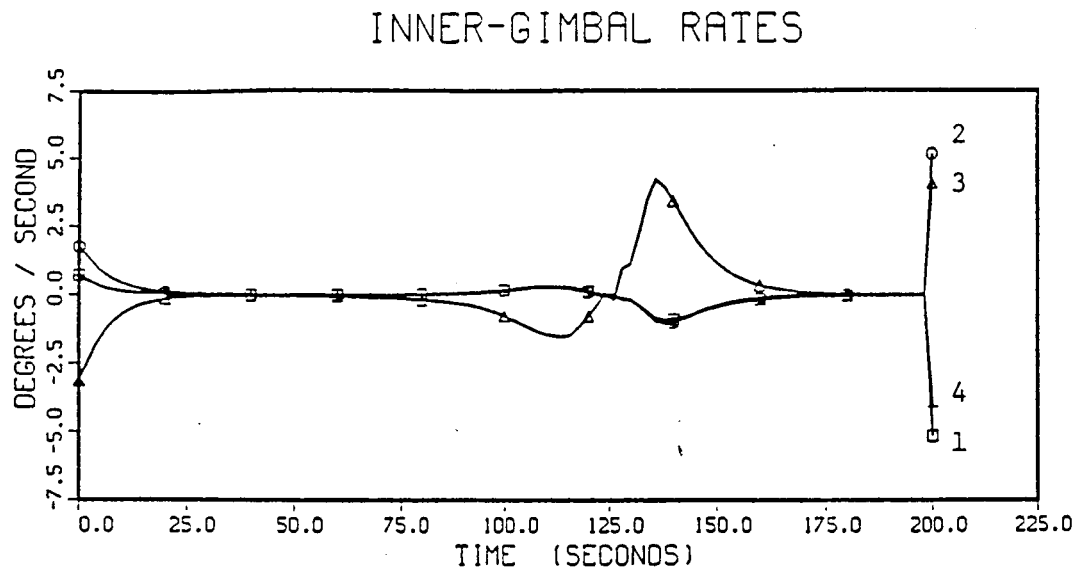


Figure 8. Gain Function during Momentum Saturation along Roll, 2-norm (674).

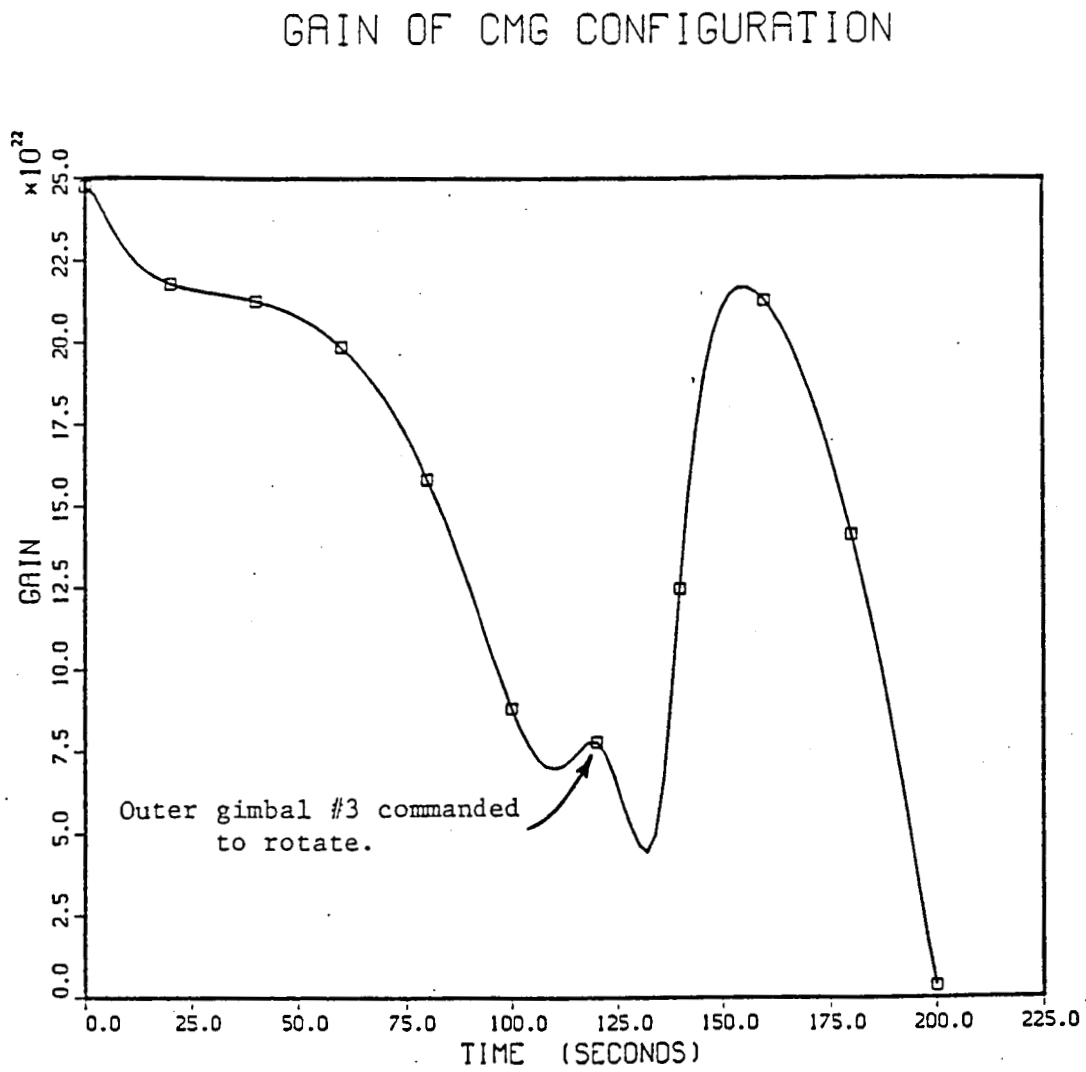


Figure 9. Gimbal Angles; Rate Change along Pitch, 2-norm (686).

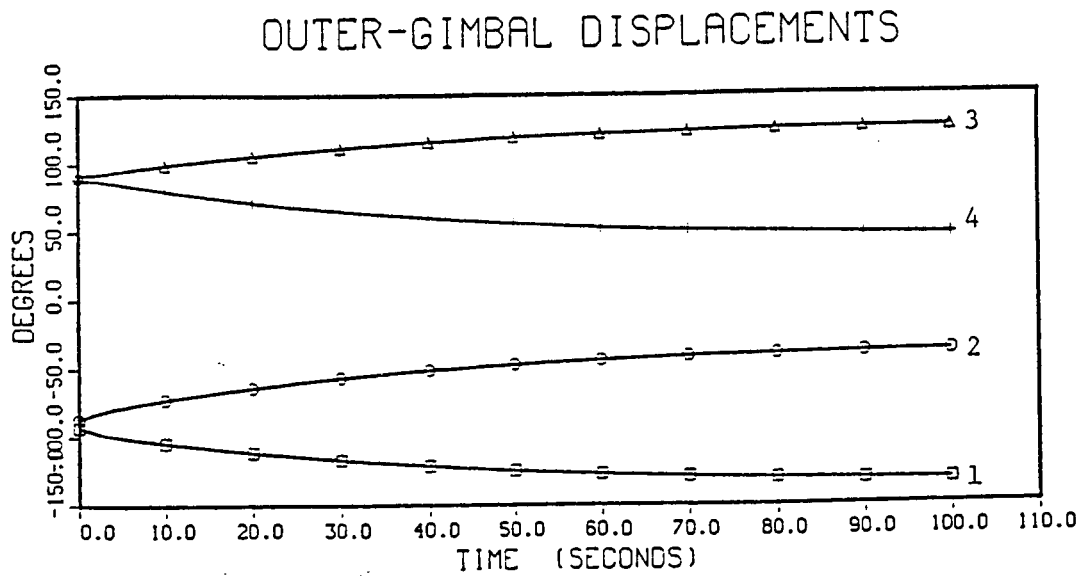
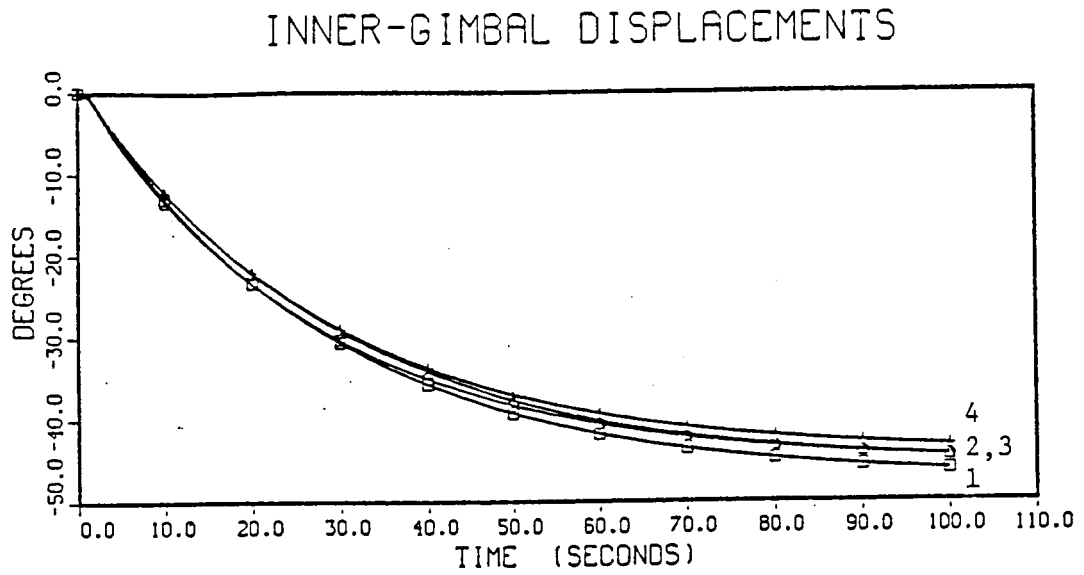


Figure 10. Gain Function during Momentum Saturation along Pitch, 2-norm (686).

